CLAIMS

What is claimed is:

1. A context-aware tokenizer comprising:

at least one context automaton module that generates a context record associated with tokens of an input data stream;

a tokenizing automaton module having a token automaton that partitions said input data stream into predefined tokens based on pattern information contained in said token automaton while simultaneously verifying contextual appropriateness based on said context record.

2. The tokenizer of claim 1 wherein said context automaton module comprises a left context automaton that populates said context record based on identified patterns that precede a given token and a right context automaton that populates said context record based on identified patterns that follow said given token.

3. The tokenizer of claim 1 wherein said tokenizing automaton module maintains a data store of predefined token classes and assigns each token identified to at least one of said predefined token classes.

4. The tokenizer of claim 3 wherein said tokenizer reports information indicative of the position and class membership of tokens identified.

5. The tokenizer of claim 1 wherein said tokenizing automaton defines a failure state, and wherein said tokenizing automaton module monitors the occurrence of said failure state to maintain a record of the longest match found involving said failure state to detect a default token in the absence of any matching patterns taken from said context automaton module and said token automaton module.

6. The tokenizer of claim 1 wherein said context automaton scans said input data stream in a left-to-right direction to acquire left context information and in a right-to-left direction to acquire right context information.

7. The tokenizer of claim 1 wherein said context automaton and said tokenizing automaton collectively obey a linear time operating constraint.

8. A text-to-speech synthesizer according to claim 1 wherein said input data stream is a text string and said tokenizing automaton module partitions said text string to include token class membership information from which the pronunciation of said text string by said synthesizer is influenced.

10. A text processor according to claim 1 wherein said input data stream comprises text and said tokenizing automaton is coupled to said text processor and operates upon said text to identify and label multi-word phrases for single unit treatment by said text processor based on information extracted by said context automaton.

11. A text processor according to claim 1 wherein said input data stream lacks word unit separation symbols and wherein said tokenizing automaton module is coupled to said text processor and operates upon said text to identify and label word units for single unit treatment by said text processor based on information extracted by said context and token automata.

12. A method of tokenizing an input stream comprising:

using at least one context automaton to generate a context record associated with tokens of said input stream;

using at least one tokenizing automaton to segment said input stream into predefined tokens based on pattern information contained in said context record.

13. The method of claim 12 wherein said step of generating said context record is performed using a left context automaton to populate said context record based on identified patterns that precede a given token and a right context automaton to populate said context record based on identified patterns that follow said given token.

14. The method of claim 12 further comprising maintaining a data store of predefined token classes and assigning each token identified to at least one of said predefined token classes.

15. The method of claim 12 further comprising reporting for each token information indicative of its position, length and class membership.

16. The method of claim 12 further comprising defining a failure state and monitoring the occurrence of said failure state to maintain a record of the longest match found involving said failure state to thereby detect a default token in the absence of any matching patterns generated by said context and token automata.

17. The method of claim 12 further comprising scanning said input stream in a first direction to acquire left context information and in a second direction to acquire right context information.

18. The method of claim 12 wherein said steps of generating a context record and of segmenting the input stream collectively obey a linear time operating constraint.

19. The method of claim 12 further comprising generating tokenization information about said input stream that includes class membership of said predefined tokens and supplying said tokenization information to a text-to-speech synthesizer.

20. The method of claim 12 further comprising generating tokenization information about said input stream that includes class membership of said predefined tokens and supplying said tokenization information to a text processor.